Package ‘drugDemand’

October 23, 2023

Type Package

Title Drug Demand Forecasting

Version 0.1.0

Date 2023-10-20

Description Performs drug demand forecasting by modeling drug dispensing data while taking into account predicted enrollment and treatment discontinuation dates. The number of skipped visits and the number of dispensed doses are modeled using zero-inflated Poisson or zero-inflated negative binomial distributions (Zeileis, Kleiber & Jackman (2008) <doi:10.18637/jss.v027.i08>) and a linear mixed-effects model (McCulloch & Searle (2001, ISBN:0-471-19364-X)), respectively.

License GPL (>= 2)

Imports Rcpp (>= 1.0.10), dplyr (>= 1.1.0), rlang (>= 1.1.0), purrr (>= 1.0.2), plotly (>= 4.10.1), survival (>= 2.41-3), mvtnorm (>= 1.1-3), ez (>= 0.4.0), stats (>= 3.5.0), icenReg (>= 2.0.15), pscl (>= 1.5.5), nlme (>= 3.1-153), eventPred (>= 0.2.0), parallel (>= 4.1.2), foreach (>= 1.5.2), doParallel (>= 1.0.17), doRNG (>= 1.8.6), tictoc (>= 1.2)

LinkingTo Rcpp

Depends R (>= 3.5.0)

LazyData true

RoxygenNote 7.2.3

Encoding UTF-8

NeedsCompilation yes

Author Kaifeng Lu [aut, cre] (<https://orcid.org/0000-0002-6160-7119>)

Maintainer Kaifeng Lu <kaifenglu@gmail.com>

Repository CRAN

Date/Publication 2023-10-23 16:10:06 UTC
drugDemand-package

R topics documented:

- drugDemand-package
- df1
- df2
- dosing_schedule_df
- drug_description_df
- f_cum_dose
- f DISPensing_models
- f_dose_new_cpp
- f_dose_ongoing_cpp
- f_dose_pp
- f_dosing_draw
- f_dosing_draw_1
- f_dosing_draw_t_1
- f_drug_demand
- f_fit_di
- f_fit_ki
- f_fit_t0
- f_fit_t1
- f_treatment_by_drug_df
- treatment_by_drug
- visitview1
- visitview2

Index 36

---

drugDemand-package    Drug Demand Forecasting

Description

Performs drug demand forecasting by modeling drug dispensing data while taking into account predicted enrollment and treatment discontinuation dates. The number of skipped visits and the number of dispensed doses are modeled using zero-inflated Poisson or zero-inflated negative binomial distributions (Zeileis, Kleiber & Jackman (2008) <doi:10.18637/jss.v027.i08>) and a linear mixed-effects model (McCulloch & Searle (2001, ISBN:0-471-19364-X)), respectively.

Details

In clinical trials, patients do not always follow protocol-specified visit and drug dispensing schedules. Patients may encounter delays in their drug dispensing appointments, skip visits altogether, or receive doses different from the protocol-specified target. Relying solely on protocol-based predictions tends to result in an overestimation of drug demand. Consequently, we propose a method that models observed drug dispensing data, thereby accounting for these deviations.

* k0 The number of skipped visits between randomization and the first drug dispensing visit.
* t0 The time elapsed between randomization and the first drug dispensing visit when k0 equals 0.
* $t_1$ The time elapsed between randomization and the first drug dispensing visit when $k_0$ is greater than 0.
* $k_i$ The number of skipped visits between two consecutive drug dispensing visits.
* $t_i$ The time elapsed between two consecutive drug dispensing visits.
* $d_i$ The number of kits dispensed at drug dispensing visits.

For $k_0$ and $k_i$, we explore several modeling options, including constant, Poisson, zero-inflated Poisson (ZIP), and zero-inflated negative binomial (ZINB) distributions.

For $t_0$, we consider various models such as constant, exponential, Weibull, log-logistic, and log-normal.

Linear regression models are applied to $t_1$ (given $k_0$) and $t_i$ (given $k_i$).

For $d_i$, we evaluate constant, linear, and linear mixed-effects models with subject random effects.

Once the dosing models are fitted to the observed drug dispensing data, we draw model parameters from their approximate posterior distributions. Subsequently, we simulate drug dispensing data after cutoff for both ongoing and new patients.

Finally, we estimate the number of kits to dispense based on the simulated data.

**Author(s)**

Kaifeng Lu, <kaifenglu@gmail.com>

**References**


The dropout indicator, where 1 corresponds to a dropout and 0 implies no dropout.

cutoff The cutoff date. For drug demand forecasting, the event of interest is treatment discontinuation. The dropout variable is set to 0 for all patients in this context.

Usage

df1

Format

An object of class tbl_df (inherits from tbl, data.frame) with 175 rows and 9 columns.

df2

The subject-level enrollment and event data after enrollment completion.

Description

A data frame with 250 rows and 9 columns:

trials The trial start date.
usubjid The unique subject ID.
randdt The randomization date for each subject.
treatment The treatment group.
treatment_description Description of the treatment group.
time The number of days elapsed since randomization.
event The event indicator, with a value of 1 indicating the occurrence of an event, and 0 indicating no event.
dropout The dropout indicator, where 1 corresponds to a dropout and 0 implies no dropout.
cutoff The cutoff date. For drug demand forecasting, the event of interest is treatment discontinuation. The dropout variable is set to 0 for all patients in this context.

Usage

df2

Format

An object of class tbl_df (inherits from tbl, data.frame) with 250 rows and 9 columns.
The dosing schedule data frame.

**Description**

A data frame with 4 rows and 4 columns:

- **drug**: The numeric code of the drug.
- **target_days**: The target number of days per treatment cycle.
- **target_kits**: The target number of kits per treatment cycle.
- **max_cycles**: The maximum number of treatment cycles.

**Usage**

dosing_schedule_df

**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 4 rows and 4 columns.

The drug description data frame.

**Description**

A data frame with 4 rows and 3 columns:

- **drug**: The numeric code of the drug.
- **drug_name**: The name of the drug.
- **dose_unit**: The dose unit for drug dispensing.

For drug demand forecasting, the default dose unit is "kit" for all drugs.

**Usage**

drug_description_df

**Format**

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 4 rows and 3 columns.
**f_cum_dose**  
*Cumulative Dose*

**Description**
Obtains the cumulative dose given treatment duration and dosing schedule.

**Usage**

\[ f_{\text{cum}}(x, w, d, N) \]

**Arguments**
- **x**: The treatment duration.
- **w**: The number of days per treatment cycle for the drug.
- **d**: The number of kits per treatment cycle for the drug.
- **N**: The maximum number of treatment cycles for the drug.

**Value**
The cumulative dose to dispense for the drug over a specified treatment duration.

**Author(s)**
Kaifeng Lu, <kaifenglu@gmail.com>

**Examples**

\[ f_{\text{cum}}(c(28, 70), 21, 2, 10000) \]

---

**f_dispensing_models**  
*Drug Dispensing Model Fit*

**Description**
Fits drug dispensing models to the observed drug dispensing data.
Usage

f_dispensing_models(
  target_days,
  vf,
  model_k0, 
  model_t0, 
  model_ki, 
  model_di, 
  nreps, 
  showplot = TRUE
)

Arguments

target_days A vector of target number of days between two drug dispensing visits by drug.
vf A data frame for subject-level drug dispensing data, including the following variables: drug, drug_name, dose_unit, usubjid, treatment, treatment_description, arrivalTime, time, event, dropout, day, dose, cum_dose, and row_id.
model_k0 The model for the number of skipped visits between randomization and the first drug dispensing visit.
model_t0 The model for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
model_ki The model for the number of skipped visits between two consecutive drug dispensing visits.
model_di The model for the dispensed doses at drug dispensing visits.
nreps The number of simulations for drawing posterior model parameters.
showplot A Boolean variable that controls whether or not to show the model fit plot. It defaults to TRUE.

Value

A list with the following components:

* common_time_model A Boolean variable that indicates whether a common time model is used for drug dispensing visits.
* fit_k0 The model fit for the number of skipped visits between randomization and the first drug dispensing visit.
* fit_t0 The model fit for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
* fit_t1 The model fit for the gap time between randomization and the first drug dispensing visit when there is visit skipping.
* fit_ki The model fit for the number of skipped visits between two consecutive drug dispensing visits.
* fit_di The model fit for the dispensed doses at drug dispensing visits.
Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
library(dplyr)

df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment,
           treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

fit <- f_dispensing_models(
  target_days = dosing_schedule_df$target_days, vf,
  model_k0 = "zip", model_t0 = "log-logistic",
  model_ki = "zip", model_di = "lme",
  nreps = 200, showplot = FALSE)

fit$fit_ki$fit_plot
```

---

**f_dose_new_cpp**  
*Dosing Date Imputation for New Patients*

Description

Imputes the dosing dates for new patients and ongoing patients with no dosing records.

Usage

```r
f_dose_new_cpp(
  usubjid,
  V,
  C,
  D,
  model_k0,
  theta_k0,
  ```
f_dose_new.cpp

model_t0,
theta_t0,
mu0,
sigma0,
model_ki,
theta_ki,
muT,
sigmaT
)

Arguments

usubjid  The unique subject ID.
V       Initialized to 0 and corresponds to the randomization visit.
C       The cutoff date relative to randomization.
D       The discontinuation date relative to randomization.
model_k0  The model for the number of skipped visits between randomization and the first drug dispensing visit.
theta_k0  The model parameters for the number of skipped visits between randomization and the first drug dispensing visit.
model_t0  The model for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
theta_t0  The model parameters for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
mu0      The regression coefficient for the linear model for the gap time between randomization and the first drug dispensing visit when there is visit skipping.
sigma0   The residual standard deviation for the linear model for the gap time between randomization and the first drug dispensing visit when there is visit skipping.
model_ki  The model for the number of skipped visits between two consecutive drug dispensing visits.
theta_ki  The model parameters for the number of skipped visits between two consecutive drug dispensing visits.
muT      The regression coefficient for the linear model for the gap time between two consecutive drug dispensing visits.
sigmaT   The residual standard deviation the linear model for the gap time between two consecutive drug dispensing visits.

Value

A data frame with two variables:
* usubjid: The unique subject ID.
* day: The dosing visit date relative to randomization.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>
Examples

```r
set.seed(529)

f_dose_new_cpp(
  usubjid = "Z001", V = 0, C = 87, D = 985,
  model_k0 = "zip", theta_k0 = c(0.6, 1.1),
  model_t0 = "log-logistic", theta_t0 = c(-1.0, 0.7),
  mu0 = 21.5, sigma0 = 1.9,
  model_ki = "zip", theta_ki = c(0.1, 0.4),
  muT = 21, sigmaT = 2.3)
```

**Description**

Imputes the dosing dates after cutoff for ongoing patients with dosing records.

**Usage**

```r
f_dose_ongoing_cpp(usubjid, V, C, D, model_ki, theta_ki, muT, sigmaT)
```

**Arguments**

- `usubjid` The unique subject ID.
- `V` The last dosing visit date relative to randomization.
- `C` The cutoff date relative to randomization.
- `D` The discontinuation date relative to randomization.
- `model_ki` The model for the number of skipped visits between two consecutive drug dispensing visits.
- `theta_ki` The model parameters for the number of skipped visits between two consecutive drug dispensing visits.
- `muT` The regression coefficient for the linear model for the gap time between two consecutive drug dispensing visits.
- `sigmaT` The residual standard deviation for the linear model for the gap time between two consecutive drug dispensing visits.

**Value**

A data frame with two variables:

- `usubjid`: The unique subject ID.
- `day`: The dosing visit date relative to randomization.
Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
set.seed(314)

f_dose_ongoing_cpp(
  usubjid = "A001", V = 297, C = 329, D = 569,
  model_ki = "zip", theta_ki = c(0.4, 2.5),
  muT = 21, sigmaT = 2.3)
```

---

**f_dose_pp**  
*Drug Demand Per Protocol*

**Description**

Obtains drug demand prediction based on protocol-assumed visit and dosing schedules.

**Usage**

```r
f_dose_pp(
  dosing_summary_t0,
  newEvents,
  treatment_by_drug_df,
  dosing_schedule_df,
  t0,
  t,
  pilevel
)
```

**Arguments**

- `dosing_summary_t0`  
  The cumulative doses dispensed before the cutoff date. It contains the following variables: `drug`, `drug_name`, `dose_unit`, and `cum_dose_t0`.

- `newEvents`  
  A data frame containing the imputed event data for both ongoing and new patients, typically obtained from the output of the `eventPred::getPrediction` function. It contains the following variables: `draw`, `usubjid`, `arrivalTime`, `treatment`, `treatment_description`, `time`, `event`, `dropout`, and `totalTime`.

- `treatment_by_drug_df`  
  A data frame indicating the treatments associated with each drug, including the following variables: `treatment`, `drug`, `drug_name`, and `dose_unit`.

- `dosing_schedule_df`  
  A data frame providing dosing schedule information. It contains the following variables: `drug`, `target_days`, `target_kits`, and `max_cycles`. 
t0 The cutoff date relative to the trial start date.
t A vector of new time points for drug dispensing predictions.
pilevel The prediction interval level.

Value
A data frame for dosing summary by drug and time point per protocol. It contains the following variables: drug, drug_name, dose_unit, t, n, pilevel, lower, upper, mean, and var.

Author(s)
Kaifeng Lu, <kaifenglu@gmail.com>

Examples

# Design stage drug demand predictions per protocol.
set.seed(312)
library(dplyr)

dosing_summary_t0 = drug_description_df %>%
dplyr::mutate(cum_dose_t0 = 0)

pred <- eventPred::getPrediction(
  df = NULL,
  to_predict = "enrollment and event",
  target_n = 250,
  target_d = 250,
  enroll_prior = list(
    model = "piecewise poisson",
    theta = c(-0.74, -1.18),
    vtheta = matrix(c(0.0087, 0, 0, 0.0082), 2, 2),
    accrualTime = c(0, 240)),
  event_prior = list(
    list(model = "log-logistic",
     theta = c(5.9, -0.2),
     vtheta = matrix(c(0.022, 0.004, 0.004, 0.012), 2, 2)),
    list(model = "log-logistic",
     theta = c(5.6, 0.02),
     vtheta = matrix(c(0.032, 0.003, 0.003, 0.012), 2, 2)),
    list(model = "log-logistic",
     theta = c(5.7, -0.3),
     vtheta = matrix(c(0.071, 0.013, 0.013, 0.054), 2, 2)))
  dropout_prior = NULL,
pilevel = 0.95,
  nyears = 3,
nreps = 200,
  showsummary = FALSE,
  showplot = FALSE,
  by_treatment = TRUE,
f_dosing_draw

```r
ngroups = 3,
alloc = c(2, 2, 1),
treatment_label = c("Drug A + Drug B",
                     "Drug C + Placebo",
                     "Drug A + Placebo")

newEvents <- pred$event_pred$newEvents

drug_name = drug_description_df$drug_name
dose_unit = drug_description_df$dose_unit
treatment_by_drug_df <- f_treatment_by_drug_df(
    treatment_by_drug, drug_name, dose_unit)

t0 = 1
nyears = 3
t1 = t0 + nyears*365
t = c(seq(t0, t1, 30), t1)
pilevel = 0.95

dosing_pred_pp <- f_dose_pp(
    dosing_summary_t0, newEvents,
    treatment_by_drug_df, dosing_schedule_df,
    t0, t, pilevel)

head(dosing_pred_pp)
```

f_dosing_draw

**Drug Dispensing Data Simulation**

**Description**

Simulates drug dispensing data after cutoff for both ongoing and new patients.

**Usage**

```r
f_dosing_draw(
    df, 
    vf, 
    newEvents, 
    treatment_by_drug_df, 
    common_time_model, 
    fit_k0, 
    fit_t0, 
    fit_t1, 
    fit_ki, 
    fit_ti, 
    fit_di,
```
t0,  
  t,  
  n.cores.max
)

Arguments

df A data frame for subject-level enrollment and event data, including the following variables: trialsdt, usubjid, randdt, treatment, treatment_description, time, event, dropout, and cutoffdt.

vf A data frame for subject-level drug dispensing data, including the following variables: drug, drug_name, dose_unit, usubjid, treatment, treatment_description, arrivalTime, time, event, dropout, day, dose, cum_dose, and row_id.

newEvents A data frame containing the imputed event data for both ongoing and new patients, typically obtained from the output of the eventPred::getPrediction function. It contains the following variables: draw, usubjid, arrivalTime, treatment, treatment_description, time, event, dropout, and totalTime.

treatment_by_drug_df A data frame indicating the treatments associated with each drug, including the following variables: treatment, drug, drug_name, and dose_unit.

common_time_model A Boolean variable that indicates whether a common time model is used for drug dispensing visits.

fit_k0 The model fit for the number of skipped visits between randomization and the first drug dispensing visit.

fit_t0 The model fit for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.

fit_t1 The model fit for the gap time between randomization and the first drug dispensing visit when there is visit skipping.

fit_ki The model fit for the number of skipped visits between two consecutive drug dispensing visits.

fit_ti The model fit for the gap time between two consecutive drug dispensing visits.

fit_di The model fit for the dispensed doses at drug dispensing visits.

t0 The cutoff date relative to the trial start date.

t A vector of new time points for drug dispensing predictions.

n.cores.max The maximum number of cores to use for parallel computing. The actual number of cores used will be the minimum of n.cores.max and half of the detected number of cores.

Value

A list with two components:

* dosing_subject_new A data frame containing observed and imputed subject-level dosing records for ongoing and new patients. It contains the following variables: draw, drug, drug_name, dose_unit, usubjid, day, dose, arrivalTime, treatment, treatment_description, time, and totalTime.
f_dosing_draw

A data frame providing dosing summaries by drug, future time point, and simulation draw for ongoing and new patients. It contains the following variables: drug, drug_name, dose_unit, t, draw, and total_dose_b.

Author(s)
Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
set.seed(431)
library(dplyr)

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  select(drug, drug_name, dose_unit, usubjid, treatment, treatment_description, arrivalTime, time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment, treatment_description, arrivalTime, time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity), .groups = "drop_last") %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

pred <- eventPred::getPrediction(
  df = df,
  to_predict = "event only",
  target_d = 250,
  event_model = "log-logistic",
  dropout_model = "none",
  pilevel = 0.95,
  nyears = 3,
  nreps = 200,
  showsummary = FALSE,
  showplot = FALSE,
  by_treatment = TRUE)

newEvents <- pred$event_pred$newEvents

drug_name = drug_description_df$drug_name
dose_unit = drug_description_df$dose_unit
treatment_by_drug <- f_treatment_by_drug_df(treatment_by_drug_d$f_treatment_by_drug_d)
```
fit <- f_dispensing_models(
  target_days = dosing_schedule_df$target_days, vf,
  model_k0 = "zip", model_t0 = "log-logistic",
  model_ki = "zip", model_di = "lme",
  nreps = 200, showplot = FALSE)

trialsdt = df$trialsdt[1]
cutoffdt = df$cutoffdt[1]
t0 = as.numeric(cutoffdt - trialsdt + 1)
nyears = 3
t1 = t0 + nyears*365
t = c(seq(t0, t1, 30), t1)

a <- f_dosing_draw(
  df, vf, newEvents, treatment_by_drug_df,
  fit$common_time_model, fit$fit_k0, fit$fit_t0, fit$fit_t1,
  fit$fit_ki, fit$fit_ti, fit$fit_di, t0, t,
  n.cores.max = 2)

head(a$dosing_subject_new)
head(a$dosing_summary_new)
Arguments

- **i**
  - The iteration number.
- **common_time_model**
  - A Boolean variable that indicates whether a common time model is used for drug dispensing visits.
- **fit_k0**
  - The model fit for the number of skipped visits between randomization and the first drug dispensing visit.
- **fit_t0**
  - The model fit for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
- **fit_t1**
  - The model fit for the gap time between randomization and the first drug dispensing visit when there is visit skipping.
- **fit_ki**
  - The model fit for the number of skipped visits between two consecutive drug dispensing visits.
- **fit_ti**
  - The model fit for the gap time between two consecutive drug dispensing visits.
- **fit_di**
  - The model fit for the dispensed doses at drug dispensing visits.
- **vf_ongoing**
  - The observed drug dispensing data for ongoing patients.
- **vf_new**
  - The randomization date for new patients and ongoing patients with no drug dispensing records.
- **vf_ongoing1**
  - The last observed drug dispensing date for ongoing patients, with or without the associated drug information.
- **vf_new1**
  - The randomization date for new patients and ongoing patients with no drug dispensing records, with or without the associated drug information.
- **treatment_by_drug_df**
  - A data frame indicating the treatments associated with each drug, including the following variables: treatment, drug, drug_name, and dose_unit.
- **t**
  - A vector of new time points for drug dispensing predictions.

Value

- A list of two components:
  - *dosing_subject_newi* for the drug dispensing data at the subject level by date for the given iteration.
  - *dosing_summary_newi* for the drug dispensing summary data by drug, time, and simulation draw for the given iteration.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
set.seed(431)
library(dplyr)
```
df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment, 
         treatment_description, arrivalTime, 
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment, 
           treatment_description, arrivalTime, 
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

pred <- eventPred::getPrediction(
  df = df,
  to_predict = "event only",
  target_d = 250,
  event_model = "log-logistic",
  dropout_model = "none",
  pilevel = 0.95,
  nye = 3,
  nreps = 200,
  showsummary = FALSE,
  showplot = FALSE,
  by_treatment = TRUE)
newEvents <- pred$event_pred$newEvents

drug_name = drug_description_df$drug_name

dose_unit = drug_description_df$dose_unit

treatment_by_drug_df <- f_treatment_by_drug_df(
  treatment_by_drug, drug_name, dose_unit)

fit <- f_dispensing_models(
  target_days = dosing_schedule_df$target_days, vf, 
  model_k0 = "zip", model_t0 = "log-logistic", 
  model_ki = "zip", model_di = "lme", 
  nreps = 200, showplot = FALSE)

trialsdt = df$trialsdt[1]
cutoffdt = df$cutoffdt[1]
t0 = as.numeric(cutoffdt - trialsdt + 1)
nyears = 3

t1 = t0 + nyears*365

l = length(unique(newEvents$draw))
l = length(unique(treatment_by_drug_df$drug))
# all ongoing subjects
def_unames1 <- df %>% filter(event == 0)
unames1 <- df_unames1$usubjid

# ongoing subjects with dosing records
def_unames2 <- vf %>% filter(event == 0) %>%
  group_by(usubjid) %>% slice(n())
unames2 <- df_unames2$usubjid

### dosing data for ongoing patients ###
vf1 <- vf %>%
  filter(usubjid %in% unames2) %>%
  select(drug, drug_name, dose_unit, usubjid, day, dose)

# replicate for nreps times
vf1_rep = tibble(draw = 1:nreps) %>% cross_join(vf1)

df1 <- newEvents %>%
  filter(usubjid %in% unames1) %>% select(-c(event, dropout))

vf_ongoing <- vf1_rep %>%
  inner_join(df1, by = c("draw", "usubjid"))

### new patients and ongoing patients with no dosing records ###
df_new <- newEvents %>%
  filter(arrivalTime > t0 | usubjid %in% setdiff(unames1, unames2))

vf_new <- purrr::map_dfr(1:l, function(h) {
  df_new %>%
    inner_join(treatment_by_drug_df %>% filter(drug == h),
      by = "treatment") %>%
    select(-c(event, dropout))
})

# only keep the last record for each patient in each draw
vf_ongoing1 <- vf_ongoing %>%
  group_by(draw, usubjid) %>% slice(n()) %>%
  mutate(V = day - 1,
    C = as.numeric(t0 - arrivalTime),
    D = pmin(time - 1, t1 - arrivalTime)) %>%
  select(-c(drug, drug_name, dose_unit, day, dose))

### new patients and ongoing patients with no dosing records ###
vf_new1 <- vf_new %>%
  group_by(draw, usubjid) %>% slice(n()) %>%
  mutate(V = 0,
    C = as.numeric(t0 - arrivalTime),
    D = pmin(time - 1, t1 - arrivalTime)) %>%
  select(-c(drug, drug_name, dose_unit))

# first iteration to extract subject and summary data
list1 <- f_dosing_draw_t_1(
  i, fit$common_time_model, 
  fit$fit_k0, fit$fit_t0, fit$fit_t1, 
  fit$fit_ki, fit$fit_ti, fit$fit_di, 
  vf_ongoing, vf_new, vf_ongoing1, vf_new1, 
  treatment_by_drug_df, t)

head(list1$dosing_subject_newi)
head(list1$dosing_summary_newi)

---

**f_dosing_draw_t_1**  
*Drug Dispensing Visit Dates for One Iteration*

**Description**

Obtains drug dispensing visit dates for one iteration.

**Usage**

```r
f_dosing_draw_t_1(
  i, 
  fit_k0, 
  fit_t0, 
  fit_t1, 
  fit_ki, 
  fit_ti, 
  vf_ongoing1, 
  vf_new1 
)
```

**Arguments**

- **i**: The iteration number.
- **fit_k0**: The model fit for the number of skipped visits between randomization and the first drug dispensing visit.
- **fit_t0**: The model fit for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.
- **fit_t1**: The model fit for the gap time between randomization and the first drug dispensing visit when there is visit skipping.
- **fit_ki**: The model fit for the number of skipped visits between two consecutive drug dispensing visits.
- **fit_ti**: The model fit for the gap time between two consecutive drug dispensing visits.
- **vf_ongoing1**: The last observed drug dispensing date for ongoing patients, with or without the associated drug information.
- **vf_new1**: The randomization date for new patients and ongoing patients with no drug dispensing records, with or without the associated drug information.
Value

Drug dispensing visit dates at the subject level.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
data.set.seed(431)
library(dplyr)
df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid,
           treatment, treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())
pred <- eventPred::getPrediction(
  df = df,
  to_predict = "event only",
  target_d = 250,
  event_model = "log-logistic",
  dropout_model = "none",
  pilevel = 0.95,
  nyears = 3,
  nreps = 200,
  showsummary = FALSE,
  showplot = FALSE,
  by_treatment = TRUE)
newEvents <- pred$event_pred$newEvents
drug_name = drug_description_df$drug_name
dose_unit = drug_description_df$dose_unit
treatment_by_drug_df <- f_treatment_by_drug_df(
  treatment_by_drug, drug_name, dose_unit)
fit <- f_dispensing_models(
```
target_days = dosing_schedule_df$target_days, vf,
model_k0 = "zip", model_t0 = "log-logistic",
model_ki = "zip", model_di = "lme",
nreps = 200, showplot = FALSE)

trialsdt = df$trialsdt[1]
cutoffdt = df$cutoffdt[1]
t0 = as.numeric(cutoffdt - trialsdt + 1)
years = 3
$t_1 = t0 + years*365$
t = c(seq(t0, t1, 30), t1)

nreps = length(unique(newEvents$draw))
l = length(unique(treatment_by_drug_df$drug))

# all ongoing subjects
df_unames1 <- df %>% filter(event == 0)
unames1 <- df_unames1$usubjid

# ongoing subjects with dosing records
df_unames2 <- vf %>% filter(event == 0) %>%
  group_by(usubjid) %>% slice(n())
unames2 <- df_unames2$usubjid

### dosing data for ongoing patients ###
vf1 <- vf %>%
  filter(usubjid %in% unames2) %>%
  select(drug, drug_name, dose_unit, usubjid, day, dose)

# replicate for nreps times
vf1_rep = tibble(draw = 1:nreps) %>% cross_join(vf1)

df1 <- newEvents %>%
  filter(usubjid %in% unames1) %>%
  select(-c(event, dropout))

vf_ongoing <- vf1_rep %>%
  inner_join(df1, by = c("draw", "usubjid"))

### new patients and ongoing patients with no dosing records ###
df_new <- newEvents %>%
  filter(arrivalTime > t0 | usubjid %in% setdiff(unames1, unames2))

vf_new <- purrr::map_dfr(1:l, function(h) {
  df_new %>%
    inner_join(treatment_by_drug_df %>% filter(drug == h),
      by = "treatment") %>%
    select(-c(event, dropout))})

# only keep the last record for each patient in each draw
vf_ongoing1 <- vf_ongoing %>%
  group_by(draw, usubjid) %>% slice(n()) %>%
### f_drug_demand

**Drug Demand Prediction**

**Description**

Obtains drug demand prediction via modeling and simulation.

**Usage**

```r
f_drug_demand(
  df = NULL,
  newEvents = NULL,
  visitview = NULL,
  drug_description_df = NULL,
  treatment_by_drug = NULL,
  dosing_schedule_df = NULL,
  model_k0 = "zip",
  model_t0 = "exponential",
  model_ki = "zip",
  model_di = "lme",
  pilevel = 0.9,
  nyears = 4,
  nreps = 500,
  n.cores.max = 10,
  showplot = TRUE
)
```
Arguments

**df**
A data frame for subject-level enrollment and event data, including the following variables: trialsdt, usubjid, randdt, treatment, treatment_description, time, event, dropout, and cutoffdt.

**newEvents**
A data frame containing the imputed event data for both ongoing and new patients, typically obtained from the output of the `eventPred::getPrediction` function. It contains the following variables: draw, usubjid, arrivalTime, treatment, treatment_description, time, event, dropout, and totalTime.

**visitview**
A data frame containing the observed drug dispensing data, including the following variables: usubjid, visit, date, drug, drug_name, dose_unit, kit_number, and dispensed_quantity.

**drug_description_df**
The drug description data frame including drug, drug_name, and dose_unit. It must be specified at the design stage. It will be replaced with the observed information at the analysis stage.

**treatment_by_drug**
The indicator matrix of treatment by drug combinations.

**dosing_schedule_df**
A data frame providing dosing schedule information. It contains the following variables: drug, target_days, target_kits, and max_cycles.

**model_k0**
The model for the number of skipped visits between randomization and the first drug dispensing visit.

**model_t0**
The model for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.

**model_ki**
The model for the number of skipped visits between two consecutive drug dispensing visits.

**model_di**
The model for the dispensed doses at drug dispensing visits.

**pilevel**
The prediction interval level.

**nyears**
The number of years after the data cut for prediction.

**nreps**
The number of replications for simulation.

**n.cores.max**
The maximum number of cores to use for parallel computing. The actual number of cores used will be the minimum of n.cores.max and half of the detected number of cores.

**showplot**
A Boolean variable that controls whether or not to show the drug dispensing model fit and drug demand prediction plots. It defaults to TRUE.

Value

A list with the following components:

* **common_time_model** A Boolean variable that indicates whether a common time model is used for drug dispensing visits.

* **fit_k0** The model fit for the number of skipped visits between randomization and the first drug dispensing visit.
* fit_t0 The model fit for the gap time between randomization and the first drug dispensing visit when there is no visit skipping.

* fit_t1 The model fit for the gap time between randomization and the first drug dispensing visit when there is visit skipping.

* fit_ki The model fit for the number of skipped visits between two consecutive drug dispensing visits.

* fit_ti The model fit for the gap time between two consecutive drug dispensing visits.

* fit_di The model fit for the dispensed doses at drug dispensing visits.

* dosing_subject A data frame for the observed and imputed subject-level dosing records.

* dosing_pred_df A data frame for dosing summary by drug and time point.

* dosing_pred_pp A data frame for dosing summary by drug and time point per protocol.

* dosing_pred_plot A plot object for dosing prediction.

**Author(s)**

Kaifeng Lu, <kaifenglu@gmail.com>

**Examples**

```r
set.seed(529)

tictoc::tic("event prediction")

pred <- eventPred::getPrediction(
  df = df2,
  to_predict = "event only",
  target_d = 250,
  event_model = "log-logistic",
  dropout_model = "none",
  pilevel = 0.95,
  nyears = 1,
  nreps = 200,
  showplot = FALSE,
  by_treatment = TRUE)

tictoc::toc()

tictoc::tic("drug demand prediction")

a <- f_drug_demand(
  df = df2,
  newEvents = pred$event_pred$newEvents,
  visitview = visitview2,
  treatment_by_drug = treatment_by_drug,
  dosing_schedule_df = dosing_schedule_df,
  model_k0 = "zip",
)
model_t0 = "log-logistic",
model_ki = "zip",
model_di = "lme",
pilevel = 0.95,
nyears = 1,
nreps = 200,
n.cores.max = 2,
showplot = FALSE)
tictoc::toc()
a$dosing_pred_plot

---

f_fit_di  Linear Mixed-Effects Model Fit for Dispensed Doses

Description
Fits a linear mixed-effects model to the dispensed doses.

Usage
f_fit_di(df, model, nreps, showplot = TRUE)

Arguments
- df: The subject-level dosing data, including usubjid, day, drug, and dose.
- model: The model used to analyze the dispensed doses, with options including "constant", "lm" (linear model), and "lme" (linear mixed-effects model).
- nreps: The number of simulations for drawing posterior model parameters.
- showplot: A Boolean variable that controls whether or not to show the residual plot. It defaults to TRUE.

Value
A list of results from the model fit, including
- * model: The specific model used in the analysis.
- * mud: The estimated mean dose.
- * vmud: The estimated variance of mud.
- * sigmab: The estimated between-subject standard deviation.
- * sigmae: The estimated within-subject residual standard deviation.
- * aic: The Akaike Information Criterion value for the model fit.
- * bic: The Bayesian Information Criterion value for the model fit.
Additionaly, the function provies:
* A residual plot.
* Posterior draws of model parameters.
* Subject random effects.

**Author(s)**

Kaifeng Lu, <kaifenglu@gmail.com>

**Examples**

```r
library(dplyr)

df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment,
           treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

vf1 <- vf %>% filter(drug == 3)
fit_di <- f_fit_di(vf1, model = "lm", nreps = 200)
```

---

**f_fit_ki**

*Count Model Fit for Number of Skipped Visits*

**Description**

Fits a count model to the number of skipped visits.

**Usage**

```r
f_fit_ki(df, model, nreps, showplot = TRUE)
```
Arguments

- **df**: The subject-level dosing data, including skipped to indicate the number of skipped visits.
- **model**: The count model used to analyze the number of skipped visits, with options including "constant", "poisson", "zip" for zero-inflated Poisson, and "zinb" for zero-inflated negative binomial.
- **nreps**: The number of simulations for drawing posterior model parameter values.
- **showplot**: A Boolean variable that controls whether or not to show the fitted count bar chart. It defaults to TRUE.

Value

A list of results from the model fit that includes:

- **model**: The specific model used in the analysis.
- **theta**: The estimated model parameters.
- **vtheta**: The estimated covariance matrix of theta.
- **aic**: The Akaike Information Criterion value for the model fit.
- **bic**: The Bayesian Information Criterion value for the model fit.

Additionally, the function provides:

- A fitted count bar chart.
- Posterior draws of model parameters.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
library(dplyr)

df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment,
           treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())
```
vf <- vf %>%
  left_join(dosing_schedule_df, by = "drug")

# time from randomization to the first drug dispensing visit
df_k0 <- vf %>%
  filter(row_id == 1) %>%
  mutate(time = day,
         skipped = floor((time - target_days/2)/target_days) + 1)

fit_k0 <- f_fit_ki(df_k0, model = "zip", nreps = 200)

---

f_fit_t0  
Time-to-Event Model Fit for Dispensing Delay After Randomization

Description

Fits a specified time-to-event model to the gap time between randomization and the first drug dispensing visit when there is no visit skipping.

Usage

f_fit_t0(df, model, nreps, showplot = TRUE)

Arguments

df  The subject-level dosing data, including the following variables:
  * time: The number of days between randomization to the first drug dispensing visit (first drug dispensing date - randomization date + 1).
  * status: The event indicator which equals 1.
  * left: Equals time - 1, used to indicate the left endpoint of an interval for interval censoring.
  * right: Equals time, used to indicate the right endpoint of an interval for interval censoring.
model  The event model used to analyze the gap time between randomization and the first drug dispensing visit when there is no visit skipping, with options including "constant", "exponential", "weibull", "log-logistic", and "log-normal".
nreps  The number of simulations for drawing posterior model parameter values.
showplot  A Boolean variable that controls whether or not to show the fitted time-to-event bar chart. It defaults to TRUE.
Value

A list of results from the model fit that includes
* model: The specific model used in the analysis.
* theta: The estimated model parameters.
* vtheta: The estimated covariance matrix of theta.
* aic: The Akaike Information Criterion value for the model fit.
* bic: The Bayesian Information Criterion value for the model fit.

Additionally, the function provides:
* A fitted time-to-event bar chart.
* Posterior draws of model parameters.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

```r
library(dplyr)

df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment,
           treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
             .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

vf <- vf %>%
  left_join(dosing_schedule_df, by = "drug")

# time from randomization to the first drug dispensing visit
df_k0 <- vf %>%
  filter(row_id == 1) %>%
  mutate(time = day,
          skipped = floor((time - target_days/2)/target_days) + 1)

df_t0 <- df_k0 %>%
  filter(skipped == 0) %>%
  ```
f_fit_ti

mutate(left = time - 1, right = time, status = 1)

fit_t0 <- f_fit_t0(df_t0, model = "log-logistic", nreps = 200)

---

**f_fit_ti  Linear Regression Model Fit for Gap Times**

**Description**

Fits a linear regression model to the gap times.

**Usage**

```r
f_fit_ti(df, model = "lm", nreps, showplot = TRUE)
```

**Arguments**

- `df`: The subject-level dosing data, including the following variables:
  - `time`: The gap time to the next drug dispensing visit.
  - `skipped`: The number of skipped visits.
  - `k1`: The covariate for the linear regression. It equals `skipped` for the gap time between randomization and the first drug dispensing visit and `skipped + 1` for the gap time between two consecutive drug dispensing visits.

- `model`: The model used to analyze the gap time. Currently, it only supports the linear model ("lm").

- `nreps`: The number of simulations for drawing posterior model parameter values.

- `showplot`: A Boolean variable that controls whether or not to show the residual plot. It defaults to `TRUE`.

**Value**

A list of results from the regression model fit, including
- `model`: The specific model used in the analysis.
- `beta`: The estimated regression coefficient for the covariate.
- `vbeta`: The estimated variance of beta.
- `sigma`: The estimated residual standard deviation.
- `df`: The residual degrees-of-freedom.
- `aic`: The Akaike Information Criterion value for the model fit.
- `bic`: The Bayesian Information Criterion value for the model fit.

Additionally, the function provides:
- A residual plot.
- Posterior draws of model parameters.
Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

library(dplyr)

df <- df2 %>%
  mutate(arrivalTime = as.numeric(randdt - trialsdt + 1))

vf <- visitview2 %>%
  inner_join(df, by = "usubjid") %>%
  mutate(day = as.numeric(date - randdt + 1)) %>%
  select(drug, drug_name, dose_unit, usubjid, treatment,
         treatment_description, arrivalTime,
         time, event, dropout, day, dispensed_quantity) %>%
  group_by(drug, drug_name, dose_unit, usubjid, treatment,
           treatment_description, arrivalTime,
           time, event, dropout, day) %>%
  summarise(dose = sum(dispensed_quantity),
            .groups = "drop_last") %>%
  mutate(cum_dose = cumsum(dose)) %>%
  group_by(drug, drug_name, dose_unit, usubjid) %>%
  mutate(row_id = row_number())

vf <- vf %>%
  left_join(dosing_schedule_df, by = "drug")

# time from randomization to the first drug dispensing visit
df_k0 <- vf %>%
  filter(row_id == 1) %>%
  mutate(time = day,
         skipped = floor((time - target_days/2)/target_days) + 1)

df_t1 <- df_k0 %>%
  filter(skipped > 0) %>%
  mutate(k1 = skipped)

fit_t1 <- f_fit_ti(df_t1, model = "lm", nreps = 200)
Usage

f_treatment_by_drug_df(treatment_by_drug, drug_name, dose_unit)

Arguments

treatment_by_drug
  The indicator matrix of treatment by drug combinations.
drug_name
  The name of the drug.
dose_unit
  The dose unit used for drug dispensing.

Value

A data frame indicating the treatments associated with each drug, including the following variables:
treatment, drug, drug_name, and dose_unit.

Author(s)

Kaifeng Lu, <kaifenglu@gmail.com>

Examples

drug_name = drug_description_df$drug_name
dose_unit = drug_description_df$dose_unit
treatment_by_drug_df <- f_treatment_by_drug_df(
  treatment_by_drug, drug_name, dose_unit)
treatment_by_drug_df

format

treatment_by_drug  The indicator matrix of treatment by drug combinations.

Description

A matrix with dimensions k x l, where k equals 3 representing the number of treatment groups, and l equals 4 representing the number of drugs. In this matrix, a value of 1 signifies the presence of the drug within a treatment group, while a value of 0 indicates the absence of the drug in that particular treatment group.

Usage

treatment_by_drug

Format

An object of class matrix (inherits from array) with 3 rows and 4 columns.
The observed subject drug dispensing data before enrollment completion.

**Description**

A data frame with 2290 rows and 8 columns:

- **usubjid**: The unique subject ID.
- **visit**: The drug dispensing visit, e.g. "Cycle 1 Day 1".
- **date**: The date of the drug dispensing visit.
- **drug**: The numeric code of the drug.
- **drug_name**: The name of the drug.
- **dose_unit**: The dose unit for drug dispensing.
- **kit_number**: The kit number for drug dispensing.
- **dispensed_quantity**: The number of kits dispensed at the visit.

**Usage**

visitview1

**Format**

An object of class tbl_df (inherits from tbl, data.frame) with 2290 rows and 8 columns.

The observed subject drug dispensing data after enrollment completion.

**Description**

A data frame with 5006 rows and 8 columns:

- **usubjid**: The unique subject ID.
- **visit**: The drug dispensing visit, e.g. "Cycle 1 Day 1".
- **date**: The date of the drug dispensing visit.
- **drug**: The numeric code of the drug.
- **drug_name**: The name of the drug.
- **dose_unit**: The dose unit for drug dispensing.
- **kit_number**: The kit number for drug dispensing.
- **dispensed_quantity**: The number of kits dispensed at the visit.
Usage

visitview2

Format

An object of class tbl_df (inherits from tbl, data.frame) with 5006 rows and 8 columns.
Index

* datasets
  df1, 3
  df2, 4
  dosing_schedule_df, 5
  drug_description_df, 5
  treatment_by_drug, 33
  visitview1, 34
  visitview2, 34

  df1, 3
  df2, 4
  dosing_schedule_df, 5
  drug_description_df, 5
  drugDemand-package, 2

  f_cum_dose, 6
  f_dispensing_models, 6
  f_dose_new_cpp, 8
  f_dose_ongoing_cpp, 10
  f_dose_pp, 11
  f_dosing_draw, 13
  f_dosing_draw_1, 16
  f_dosing_draw_t_1, 20
  f_drug_demand, 23
  f_fit_di, 26
  f_fit_ki, 27
  f_fit_t0, 29
  f_fit_ti, 31
  f_treatment_by_drug_df, 32

  treatment_by_drug, 33

  visitview1, 34
  visitview2, 34